Patent **Attorney Docket No.: Intel 2207/17039**
Assignee: Intel Corporation

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICANTS	:	Per H. HAMMARLUND et al.
SERIAL NO.	:	10/749,271
FILED	:	December 30, 2003
FOR	:	METHOD AND APPARATUS FOR ENABLING AN ADAPTIVE REPLAY LOOP IN A PROCESSOR
GROUP ART UNIT	:	2183
EXAMINER	:	Jacob Andrew PETRANEK
CUSTOMER NO.	:	25693

M/S APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

ATTENTION: Board of Patent Appeals and Interferences

APPEAL BRIEF

Sir:

This brief is in furtherance of the Notice of Appeal, filed in this case on January 8, 2007.

1. REAL PARTY IN INTEREST

The real party in interest in this matter is Intel Corporation. (Recorded October 26, 2004; Reel/Frame 015926/0181).

2. RELATED APPEALS AND INTERFERENCES

There are no related appeals.

3. STATUS OF THE CLAIMS

Claims 1-30 are pending and rejected in this application. No claims are allowed, withdrawn, objected to, or cancelled. The claims in their current form (including those claims under appeal) are presented in the Appendix – Section 8 – Claims on Appeal.

4. STATUS OF AMENDMENTS

The claims listed on page A-1 of the Appendix attached to this Appeal Brief reflect the present status of the claims.

5. SUMMARY OF THE CLAIMED SUBJECT MATTER

Claim 1 generally describes an embodiment adaptive replay system comprising a staging unit to forward an instruction in a replay loop parallel to an execution unit (*see e.g.*, paragraph [0029] – Figure 3, 345), a selector device coupled to said staging area to place said instruction in an optimal position within said replay loop (*see e.g.*, paragraph [0029] – Figure 3, 350), and a

scoreboard coupled to said selector device to store status information for said instruction (*see e.g.*, paragraph [0033] – Figure 3, 365).

Claim 13 generally describes an embodiment computer processing system comprising a multiplexer having a first input, a second input, and an output (*see e.g.*, paragraph [0028] – Figure 3, 325); a scheduler coupled to said multiplexer first input (*see e.g.*, paragraph [0028] – Figure 3, 320); an execution unit coupled to said multiplexer output; a memory device coupled to said execution unit (*see e.g.*, paragraph [0028] – Figure 3, 330); and a replay system having an output coupled to said second multiplexer input (*see e.g.*, paragraph [0029] – Figure 3, 340). In this embodiment, the replay system includes a staging unit coupled to said multiplexer output to forward an instruction in a replay loop parallel to an execution unit (*see e.g.*, paragraph [0029] – Figure 3, 345), a selector device coupled to said staging unit, said selector multiplexer is adapted to place an instruction to an optimal position within said replay loop, (*see e.g.*, paragraph [0029] – Figure 3, 350) and a scoreboard coupled to said selector device to store status information for said instruction (*see e.g.*, paragraph [0033] – Figure 3, 365).

Claim 23 generally describes a embodiment method a method of processing a computer instruction in a replay loop comprising: analyzing multiple instructions from a staging unit (*see e.g.*, paragraph [0046] – Figure 9, 905), checking a scoreboard for latency information for each of said multiple instructions (*see e.g.*, paragraph [0046] – Figure 9, 910), checking said scoreboard for dependency information for each of said multiple instructions (*see e.g.*, paragraph [0033]), checking said scoreboard for resource conflicts for each of said multiple instructions (*see e.g.*, paragraph [0046] – Figure 9, 915), determining an optimal position for each of said multiple instructions in

said replay loop (*see e.g.*, paragraph [0046] – Figure 9, 920), and moving each of said instructions to said optimal position in said replay loop (*see e.g.*, paragraph [0046] – Figure 9, 925).

Claim 27 generally describes an embodiment a set of instructions residing in a storage medium, said set of instructions capable of being executed by a processor to implement a method of processing a computer instruction in a replay loop comprising: analyzing multiple instructions from a staging unit (*see e.g.*, paragraph [0046] – Figure 9, 905); checking a scoreboard for latency information for each of said multiple instructions (*see e.g.*, paragraph [0046] – Figure 9, 910); checking said scoreboard for dependency information for each of said multiple instructions (*see e.g.*, paragraph [0033]); checking said scoreboard for resource conflicts for each of said multiple instructions (*see e.g.*, paragraph [0046] – Figure 9, 915); determining an optimal position for each of said multiple instructions in said replay loop (*see e.g.*, paragraph [0046] – Figure 9, 920); and moving each of said instructions to said optimal position in said replay loop (*see e.g.*, paragraph [0046] – Figure 9, 925).

6. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

A. Are claims 1-2, 4, and 7 anticipated under 35 U.S.C. §102(b) by Merchant et al., U.S. Patent No. 6,385,715 ("Merchant")?

B. Are claims 3, 9-15, 18-23, 25-27, and 29-30 rendered obvious under 35 U.S.C. §103(a) as being unpatentable over Merchant?

C. Are claims 5-6 and 16-17 rendered obvious under U.S.C. §103(a) over Merchant, in view of Merchant et al., U.S. Patent No. 6,163,838 ("Merchant '838")?

D. Are claims 8, 24, and 28 rendered obvious under U.S.C. §103(a) as being unpatentable over Merchant, in view of Topham et al, U.S. Patent No. 6,944,853 ("Topham")?

7. ARGUMENT

A. Claims 1-2, 4, and 7 are not anticipated under 35 U.S.C. §102(b) by Merchant.

Applicants respectfully submit the cited references do not teach, suggest or describe at least "an adaptive replay system comprising:... *a selector device* coupled to said staging area *to place said instruction in an optimal position within said replay loop*..." (*e.g.*, as described in claim 1).

The Examiner asserts Merchant teaches such a selector device at column 6, lines 7-25, further including citations to element 150 and 154. *See* Office Action dated 4/20/2006, paragraph 10, page 4. Applicants disagree.

Element 150 of Merchant is described as a checker, while element 154 is described as a replay queue *unloading* controller (*emphasis* added). The cited section states:

> If the checker 150 determines that the instruction has not executed properly, the instruction will then be returned to multiplexer 116 to be replayed (*i.e.*, re-executed). Each instruction to be replayed will be returned to mux 116 via one of two paths. Specifically, if the checker 150 determines that the instruction should be replayed, the Replay Queue Loading Controller 154 determines whether the instruction should be sent through a replay loop 156 including staging queues E and F, or whether the instruction should be temporarily stored in a replay queue 170 before returning to mux 116.
>
> Instructions routed via the replay loop 156 are coupled to mux 116 via line 161. Instructions can also be routed by controller 154 for temporary storage in replay queue 170 (prior to replay). The instructions stored in replay queue 170 are output or unloaded under control of replay queue unloading controller 179. The instructions output from replay queue 170 are coupled to mux 116 via line 171. The operation of replay queue 170, Replay Queue Loading Controller 154 and Replay Queue Unloading Controller 179 are described in detail below.

Applicants submit the cited section does not teach the relevant limitations; indeed, the cited section does not refer to optimally rearranging the order of instructions at all. The first sentence of the cited section introduces the "re-execut[ion]" process. Specifically, if the checker 150 determines the instruction did not execute properly, it is returned to the multiplexer 116. To do this, *a conditional determination* is made which decides to send the instruction down one of two paths. The first outcome of this *conditional determination* sends the instruction to the replay loop 156 , while the second outcome sends it temporarily to the replay queue 170 before sending it to the mux 116. Instructions stored in replay queue are sent to controller 179, and are coupled to mux 116 (described above) by line 171.

Applicants submit at the heart of the cited section is this two-outcome *conditional determination* of where to send an instruction that hasn't executed properly. However, making a conditional determination upon which one of two steps may be followed is not the same as placing an instruction in an optimal position *within a replay queue* (as described in embodiments of the present application) at all. Indeed, the cited section does not discuss *placement* of an instruction for any reason anywhere (other than re-sending the instruction as discussed above). In order to support a proper rejection, the cited section must describe at least placing an instruction in an optimal position with a replay loop. The Merchant reference does not. Applicants submit for at least this reason, this section is inadequate to support a proper rejection of independent claim 1.

The Examiner further argues element 150 determines if a replay is necessary and element 154 determines which replay path will be used for the instruction. Therefore, it asserts, element 154 places the instruction in an optimal position for the processor by either sending it through

the staging queues E and F, or sending it to the replay queue where it could be delayed. *See* Office Action 9/6/2006, paragraph 42. Applicants maintain this assertion fails to support a proper rejection for reasons similar to those discussed above.

Specifically, placing an instruction in an optimal position *for the processor* is not the same as placing an instruction in an optimal position *within said replay loop (e.g.*, as described in embodiment of the present application). As discussed above, the cited reference makes a conditional determination entailing either returning an instruction to the replay loop 156 to be repeated or sending it temporarily to the replay queue 170 before sending it to the mux 116. Such a determination or operation entails either repeating the instruction as a whole, or sending it temporarily to a replay queue 170. It does not entail placing an instruction in an optimal position *within said replay loop (e.g.*, as described in embodiment of the present application).

Applicants maintain in order to support a proper rejection, the cited reference must describe at least placing an instruction in an optimal position *within a replay loop*. Since the for at least the reasons described above, the Merchant reference fails to do so, the current rejection of claims 1-2, 4, and 7 is lacking and should be withdrawn.

B. Claims 3, 9-15, 18-23, 25-27, and 29-30 are <u>not</u> rendered obvious under 35 U.S.C. §103(a) over Merchant.

For at least the reasons described above, Merchant fails to at least the relevant features of the claimed embodiments discussed above. As such, the current §103(a) rejection fails for similar reasons to those described above. Since, for at least the reasons described above, the current rejection fails to support a proper §103(a) rejection of claims 3, 9-15, 18-23, 25-27, and

29-30, Applicants submit the claims are allowable and the current §103(a) rejection should be withdrawn.

C. Are claims 5-6 and 16-17 rendered obvious under U.S.C. §103(a) over Merchant, in view of Merchant et al., U.S. Patent No. 6,163,838 ("Merchant '838")?

Merchant '838 fails to make up for the deficiencies of Merchant '715. Merchant '838 is also directed toward a replay system for the purpose of replaying instructions, and similar to above fails to describe at least a selector device's ability to *place an instruction in an optimal position within a replay loop* anywhere as described in embodiments of the present application. As such, the §103(a) rejection fails for similar reasons to those described above. Since, for at least the reasons described above, claims 5-6 and 16-17 depend from allowable base claims, Applicants submit they are allowable as well and the current §103(a) rejection should be withdrawn.

D. Are claims 8, 24, and 28 rendered obvious under U.S.C. §103(a) as being unpatentable over Merchant, in view of Topham et al, U.S. Patent No. 6,944,853 ("Topham")?

Topham fails to make up for the deficiencies of Merchant and Merchant '715 as well. Topham is directed toward the predicated execution of instructions in processors, but does not describe at least a selector device's ability to *place an instruction in an optimal position within a replay loop* anywhere as described in embodiments of the present application. As such, the current rejection fails for similar reasons to those described above. Since, for at least the reasons

described above, claims 8, 24, and 28 depend from allowable base claims, Applicants submit they are allowable as well and the current §103(a) rejection should be withdrawn.

CONCLUSION

Therefore, since for at least the preceding reasons each and every limitation is not taught or suggested in the cited references, Applicants submit they are inadequate to support proper 35 U.S.C. §102(b) and §103(a) rejections, and independent claim 1 should be allowed. Independent claims 13, 23, and 27 contain similar allowable limitations. Claims 2-12, 14-22, 24-26, and 28-30 depend from allowable independent claims and therefore are allowable as well.

Appellant therefore respectfully requests that the Board of Patent Appeals and Interferences reverse the Examiner's decision rejecting claims 1-30 and direct the Examiner to pass the case to issue.

The Examiner is hereby authorized to charge any additional fees which may be necessary for consideration of this paper to Kenyon & Kenyon LLP Deposit Account No. 11-0600.

Respectfully submitted,

KENYON & KENYON LLP

Dated September 17, 2007

By: /Sumit Bhattacharya/
Sumit Bhattacharya
Reg. No. 51,469

Kenyon & Kenyon LLP
333 West San Carlos Street, Suite 600
San Jose, California 95110
Tel: 408.975.7500
Fax: 408.975.7501

108803.1

APPENDIX

(Brief of Appellants Per H. Hammarlund et al.
U.S. Patent Application Serial No. 10/749,271)

8. CLAIMS ON APPEAL

1. An adaptive replay system comprising:

a staging unit to forward an instruction in a replay loop parallel to an execution unit;

a selector device coupled to said staging area to place said instruction in an optimal position within said replay loop; and

a scoreboard coupled to said selector device to store status information for said instruction.

2. The system of claim 1 wherein said staging unit is comprised of multiple stages.

3. The system of claim 1 wherein said status information is latency, dependency and resource conflict information.

4. The system of claim 2 wherein said multiple stages are equivalent in number to a number of stages in said execution unit.

5. The system of claim 2 wherein said adaptive replay system is implemented within a multiple channel processor.

6. The system of claim 5 wherein said selector device is to place said instruction in said optimal position within said replay loop, from a first channel to a second channel, based on status information for said instruction stored in said scoreboard.

7. The system of claim 1 wherein said selector device is to analyze at least one instruction per clock cycle to determine whether said at least one instruction has executed correctly.

8. The system of claim 7 wherein said selector device analyzes 3 instructions per clock cycle.

9. The system of claim 1 wherein said selector device places said instruction in said optimal position within said replay loop based on status information for said instruction stored in said scoreboard.

10. The system of claim 9 wherein said selector device can move instructions at least one position relative to a current position to said optimal position in said replay loop.

11. The system of claim 3 wherein said scoreboard stores latency and dependency information for said instruction when said instruction is first scheduled, and updates latency and dependency information for said instruction when said instruction is executed.

12. The system of claim 3 wherein said scoreboard stores resource conflicts for said instruction when said instruction encounters a resource conflict during execution.

13. A computer processing system comprising:

a multiplexer having a first input, a second input, and an output;

a scheduler coupled to said multiplexer first input;

an execution unit coupled to said multiplexer output;

a memory device coupled to said execution unit; and

a replay system having an output coupled to said second multiplexer input;

wherein said replay system includes:

a staging unit coupled to said multiplexer output to forward an instruction in a replay loop parallel to an execution unit; and

a selector device coupled to said staging unit, said selector multiplexer is adapted to place an instruction to an optimal position within said replay loop; and

a scoreboard coupled to said selector device to store status information for said instruction.

14. The system of claim 13 wherein said staging unit is comprised of multiple stages.

15. The system of claim 13 wherein said status information is latency, dependency and resource conflict information.

16. The system of claim 14 wherein said replay system is implemented within a multiple channel processor.

17. The system of claim 16 wherein said selector device is to place said instruction in said optimal position within said replay loop, from a first channel to a second channel, based on status information for said instruction stored in said scoreboard.

18. The system of claim 13 wherein said selector device is to analyze at least one instruction per clock cycle to determine whether said at least one instruction has executed correctly.

19. The system of claim 13 wherein said selector device is to place said instruction in said optimal position within said replay loop based on status information for said instruction stored in said scoreboard.

20. The system of claim 19 wherein said selector device can move instructions at least one position relative to a current position to said optimal position in said replay loop.

21. The system of claim 15 wherein said scoreboard is to store latency and dependency information for said instruction when said instruction is first scheduled, and updates latency and dependency information for said instruction when said instruction is executed.

22. The system of claim 15 wherein said scoreboard stores resource conflicts for said instruction when said instruction encounters a resource conflict during execution.

23. A method of processing a computer instruction in a replay loop comprising:

analyzing multiple instructions from a staging unit;

checking a scoreboard for latency information for each of said multiple instructions;

checking said scoreboard for dependency information for each of said multiple instructions;

checking said scoreboard for resource conflicts for each of said multiple instructions;

determining an optimal position for each of said multiple instructions in said replay loop; and

moving each of said instructions to said optimal position in said replay loop.

24. The method of claim 23 wherein analyzing multiple instructions from a staging unit, a replay selector device analyzes 3 instructions per clock cycle.

25. The method of claim 23 wherein determining an optimal position for each of said multiple instructions in said replay loop is based on latency, dependency and resource conflict information for said instruction stored in said scoreboard.

26. The method of claim 23 wherein moving each of said instructions to said optimal position in said replay loop, a replay selector device can move instructions at least one position relative to a current position to said optimal position in said replay loop.

27. A set of instructions residing in a storage medium, said set of instructions capable of being executed by a processor to implement a method of processing a computer instruction in a replay loop comprising:

analyzing multiple instructions from a staging unit;

checking a scoreboard for latency information for each of said multiple instructions;

checking said scoreboard for dependency information for each of said multiple instructions;

checking said scoreboard for resource conflicts for each of said multiple instructions;

determining an optimal position for each of said multiple instructions in said replay loop; and

moving each of said instructions to said optimal position in said replay loop.

28. The set of instructions of claim 27 wherein analyzing multiple instructions from a staging unit, a replay selector device analyzes 3 instructions per clock cycle.

29. The set of instructions of claim 27 wherein determining an optimal position for each of said multiple instructions in said replay loop is based on latency, dependency and resource conflict information for said instruction stored in said scoreboard.

30. The set of instructions of claim 27 wherein moving each of said instructions to said optimal position in said replay loop, a replay selector device can move instructions at least one position relative to a current position to said optimal position in said replay loop.

9. EVIDENCE APPENDIX

No further evidence has been submitted with this Appeal Brief.

10. RELATED PROCEEDINGS APPENDIX

Per Section 2 above, there are no related proceedings to the present Appeal.